

Abstract

Common approximation tools return low-order approximations in the vicinities of singularities. Most prior works solve this problem for univariate functions. In this work we introduce a method for approximating non-smooth multivariate functions of the form $f = g + r_+$ where $g, r \in C^{M+1}(\mathbb{R}^n)$ and the function r_+ is defined by

$$r_+(y) = \begin{cases} r(y), & r(y) \geq 0 \\ 0, & r(y) < 0 \end{cases}, \quad \forall y \in \mathbb{R}^n.$$

Given scattered (or uniform) data points $X \subset \mathbb{R}^n$, we investigate approximation by quasi-interpolation. We design a correction term, such that the corrected approximation achieves full approximation order on the entire domain. We also show that the correction term is the solution to a Moving Least Squares (MLS) problem, and as such can both be easily computed and is smooth. Last, we prove that the suggested method includes a high-order approximation to the locations of the singularities.

High order approximation to non-smooth multivariate functions

Anat Amir* David Levin*

October 11, 2016

1 Introduction

Approximation of non-smooth functions is a complicated problem. Common approximation tools, such as splines or approximations based on Fourier transform, return smooth approximations, thus relying on the smoothness of the original function for the approximation to be correct. However, the need to approximate non-smooth functions exists in many applications. For a high-order approximation of non-smooth functions, we need to allow our approximation to be non-smooth. Otherwise, in the vicinities of the singularities, we will get a low-order approximation. In this work we will suggest a method that will allow us to properly approximate non-smooth functions of a given model.

We will concentrate on functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ which may be modelled as $f = g + r_+$ where $g, r \in C^{M+1}(\mathbb{R}^n)$ and the function r_+ is defined by

$$r_+(y) = \begin{cases} r(y), & r(y) \geq 0 \\ 0, & r(y) < 0 \end{cases}, \quad \forall y \in \mathbb{R}^n.$$

Such functions are obviously continuous, but are non-smooth across the hypersurface

$$\Gamma_r := \{z \in \mathbb{R}^n : r(z) = 0\}.$$

As an example for such functions, consider shock waves, which are solutions of non-linear hyperbolic PDEs [12]. Another example would be a signed distance function [13], where the distance is measured from a disconnected set. Our goal is to achieve high-order approximations of such functions. To achieve that we will concentrate on a specific family of approximation tools.

Consider a quasi-interpolation operator Q [16]. Such an operator receives the values of a function $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ on a set of data points $X \subset \mathbb{R}^n$. The quasi-interpolation operator Q returns an approximation defined by

$$Q\phi(y) := \sum_{x \in X} q_x(y)\phi(x), \quad \forall y \in \mathbb{R}^n,$$

where $\{q_x\}$ are the quasi-interpolation basis functions, each is smooth and has compact support.

*School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv 6997801, Israel

Let h be the fill distance of X ,

$$h := \min \{L : B_L(y) \cap X \neq \emptyset, \forall y \in \mathbb{R}^n\},$$

where $B_r(y)$ is the ball of radius r centred at y . Denote

$$\Upsilon_M := \min \{L > 0 : \forall y \in \mathbb{R}^n, B_{Lh}(y) \cap X \text{ is uni-solvent for } \Pi_M(\mathbb{R}^n)\}.$$

Here,

$$\Pi_M(\mathbb{R}^n) := \{p : \mathbb{R}^n \rightarrow \mathbb{R} : \deg(p) \leq M\},$$

and $\deg(p)$ is the total degree of the polynomial p . Thus, h is the minimal radius which is guaranteed to contain a data point, and $\Upsilon_M \cdot h$ is the minimal radius that guarantees enough data points to uniquely determine each polynomial in $\Pi_M(\mathbb{R}^n)$. We will also assume that there exists $N > 0$ such that for all $y \in \mathbb{R}^n$ we have

$$\frac{\#(X \cap B_h(y))}{h^n} \leq N.$$

That is, the data set X has no accumulation points. Denote

$$R := \min \{\rho > 0 : \text{supp}(q_x) \subseteq B_{\rho h}(x), \forall x \in X\} \quad (1)$$

We assume that the operator Q has a bounded Lebesgue constant

$$L_1 := \sup \left\{ \sum_{x \in X} |q_x(y)| : y \in \mathbb{R}^n \right\} \quad (2)$$

and reproduces polynomials in $\Pi_M(\mathbb{R}^n)$. Then, the error in the quasi-interpolation,

$$E\phi := \phi - Q\phi$$

satisfies for all $\phi \in C^{M+1}(\mathbb{R}^n)$ and $y \in \mathbb{R}^n$

$$|E\phi(y)| \leq C_1 \cdot \|\phi\|_{C^{M+1}} \cdot h^{M+1}$$

where

$$C_1 = (1 + L_1) \cdot R^{M+1}$$

and

$$\|\phi\|_{C^{M+1}} := \sum_{|\beta|=M+1} \frac{\|D^\beta \phi\|_\infty}{\beta!}$$

with β a multi-index and $\|\cdot\|_\infty$ the maximum norm. That is, the operator Q has full approximation order for smooth functions [16]. On the other hand, since the approximation $Q\phi$ is always smooth, the operator gives low-order approximations in the vicinities of singularities.

One example of a quasi-interpolation operator is the MLS approximation [7, 9]. Given a function $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ and a point $y \in \mathbb{R}^n$ the MLS approximation is defined as $Q\phi(y) := p_y(y)$ where

$$p_y := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in X} \eta \left(\frac{\|y - x\|}{h} \right) \cdot (p(x) - \phi(x))^2 \quad (3)$$

Here η is a smooth weight function with compact support. The MLS approximation essentially returns the value at the point y of the M -th degree polynomial which gives the best approximation to ϕ at the data points

$$X \cap \text{supp} \left(\omega \left(\frac{\|\cdot - y\|}{h} \right) \right) .$$

This approximation is especially important in this work, since we base much of our results on our ability to adapt the MLS approximation to our non-smooth scenario.

Our initial goal for this work was to generalize the work done by Lipman and Levin [10]. In that work, the authors address the problem of approximation of univariate functions of the form

$$f(x) = g(x) + \sum_{j=1}^M \frac{\Delta_j}{j!} (x - s)_+^j ,$$

where $g \in C^{M+1}(\mathbb{R})$ and $(x - s)_+^j = \begin{cases} (x - s)^j, & x \geq s \\ 0, & x < s \end{cases}$. Indeed, such functions are continuous but not smooth. In [10], the univariate case is solved by modelling the error terms of the approximation by a quasi-interpolate Q . That is, one searches for variables $s^*, \bar{\Delta}^*$, such that the errors in the quasi-interpolation approximation of the term

$$\tilde{r}_{\bar{\Delta}^*, s^*}(x) := \sum_{j=1}^M \frac{\bar{\Delta}_j^*}{j!} (x - s^*)_+^j$$

give the best Least-squares approximation to the errors of the function f at the data points. It is shown that by adding the error of the approximation of the new term $\tilde{r}_{\bar{\Delta}^*, s^*}$ to the approximation Qf , full approximation order for the function f is achieved.

Another approach to this problem was proposed by Harten [8]. The author introduces the essentially non-oscillatory (ENO) and the subcell resolution (SR) schemes. The ENO scheme bases the approximation at each point on only some of the data points in its vicinity. Thus, disregarding points from the other side of the singularity which contaminate the approximation. The SR scheme locates the singularities by intersecting polynomials from supposedly different sides of the singularities. For an examination of these methods for univariate functions with a jump discontinuity in the derivative see [1].

Archibald et al ([2], [3]) suggest using polynomial annihilation to locate the singularity Γ_r . Of-course, once the singularity is known we can approximate each connected component of $\mathbb{R}^n \setminus \Gamma_r$ independent of the values in the other connected components.

Other approaches were suggested by Markakis and Barack [11], where the authors revise the Lagrange interpolation formula to approximate univariate discontinuous functions, and by Plaskota et al ([14], [15]), where the authors suggest using adaptive methods for this approximation. Batenkov et al ([4], [5], [6]) address a similar problem, the reconstruction of a piecewise smooth function from its integral measurements. One disadvantage of the methods mentioned above, is that they do not easily adapt to multivariate singular functions.

Thus, the main advantage of the method we suggest in this paper is its ability to deal with the multivariate case. Indeed, our method enables us to approximate multivariate functions which have non-continuous derivatives across smooth hyper-surfaces, Γ_r . Note that while the dimension n of the domain of the function f affects the required number of data points in X and the dimension of $\Pi_M(\mathbb{R}^n)$, the correction procedure is not otherwise affected by the higher dimension.

2 Main results

As described in the introduction, our goal is to fix the approximation of the function $f = g + r_+$, where $g, r \in C^{M+1}(\mathbb{R}^n)$.

Remark 1. *The decomposition $f = g + r_+$ is not unique. Indeed,*

$$f = g + r_+ = g + r + (-r)_+ .$$

However, we only correct the approximation error, for which we do have uniqueness.

We will achieve this, following the main idea in [10], by investigating the error terms

$$Ef(y) = \underbrace{Eg(y)}_{O(h^{M+1})} + Er_+(y) = Er_+(y) + O(h^{M+1}) .$$

Definition 1 (λ -neighbourhood of Γ_r). *For $\lambda > 0$ define*

$$\mathcal{G}(\lambda) := \bigcup_{z \in \Gamma_r} B_\lambda(z) .$$

Note that if the point y is far enough from the singularity, the restriction of f to the Rh neighbourhood of y is a smooth function, hence in this case there is no need to fix the approximation. Thus, we need only correct the approximation for points in the set $\mathcal{G}(Rh)$. In the following we suggest an algorithm for bounding the set $\mathcal{G}(Rh)$. We begin by estimating whether the function r returns a positive or negative value at each data point $x \in X$. For this we will need the following definition:

Definition 2 (Partition of the data points with respect to sign). *For a set $X \subset \mathbb{R}^n$ with fill-distance h and a function r we will say that the set $P \subset X$ partitions the data points in X with respect to the sign of the function r if*

1. $\forall x \in P$ either $r(x) > 0$ or $r(x) = O(h^{M+1})$.
2. $\forall x \in X \setminus P$, either $r(x) < 0$ or $r(x) = O(h^{M+1})$.

In section 4 we will introduce an algorithm that partitions the data points in X with respect to the sign of the function r . For now, let us assume that a set \mathcal{P} which partitions the data points in X with respect to the sign of the function r is known.

Remark 2. Apparently, once we have a set \mathcal{P} which partitions the data points in X with respect to the sign of the function r , we can approximate a point where r has a positive value using only the data points in \mathcal{P} , and a point where r has negative value using only the data points in $X \setminus \mathcal{P}$. However, we predict whether r has a positive or negative value only on the data points X , and not on the entire domain. Specifically, for a point close to the singularity location, Γ_r , we can not tell whether we should approximate based on \mathcal{P} or on $X \setminus \mathcal{P}$. Hence we can not rely only on the set \mathcal{P} to fix the approximation.

Definition 3 (The set \mathcal{G}). Denote

$$\mathcal{B}(y) := B_{(R+2\Upsilon_M+2)h}(y) ,$$

and define

$$\mathcal{G} := \{y \in \mathbb{R}^n : \mathcal{B}(y) \cap \mathcal{P} \text{ and } \mathcal{B}(y) \cap (X \setminus \mathcal{P}) \text{ are uni-solvent for } \Pi_M(\mathbb{R}^n)\} .$$

Of-course,

$$\mathcal{G}(Rh) \neq \mathcal{G} ,$$

however, we can prove that

$$\mathcal{G}(Rh) \subset \mathcal{G}((R+1)h) \subset \mathcal{G} .$$

This gives us a bound on the region in which we need to fix the approximations.

Theorem 1 (The domain of the correction). *If $\nabla r(z) \neq 0$ for all $z \in \Gamma_r$, then*

$$\mathcal{G}((R+1)h) \subseteq \mathcal{G} .$$

2.1 The corrected approximation

We may now describe the corrected approximation of the function f . Pick a point $y \in \mathbb{R}^n$. As explained above, there is no need to fix the approximation outside the set \mathcal{G} . Hence, if $y \in \mathbb{R}^n \setminus \mathcal{G}$ define the corrected approximation as

$$\hat{Q}f(y) = Qf(y) .$$

Otherwise assume that $y \in \mathcal{G}$. To fix the approximation of f we need a polynomial p^y which approximates the function r locally. In section 5 we will introduce two methods that will allow us to construct the approximation p^y .

Definition 4 (Smooth M -th order approximation). *We will say that a mapping $\Psi : \mathcal{G} \rightarrow \Pi_M(\mathbb{R}^n)$ is a smooth M -th order approximation to the function $r : \mathbb{R}^n \rightarrow \mathbb{R}$ if Ψ satisfies the following conditions :*

- There exists a constant $C_2 > 0$, independent of h , such that for all $y \in \mathcal{G}$,

$$|(\Psi(y))(u) - r(u)| \leq C_2 \cdot (\|g\|_{C^{M+1}} + \|r\|_{C^{M+1}}) \cdot h^{M+1} , \forall u \in B_{Rh}(y) .$$

- Let $\{p_\alpha\}_{\alpha \in I}$ be a polynomial basis of $\Pi_M(\mathbb{R}^n)$ and write

$$\Psi(y) = \sum_{\alpha \in I} \lambda_\alpha(y) \cdot p_\alpha .$$

Then, the mappings $y \mapsto \lambda_\alpha(y)$ are infinitely smooth.

If the mappings $y \mapsto p^y$ are a smooth M -th order approximation to r , then we may define the corrected approximation as

$$\widehat{Q}f(y) = Qf(y) + E((p^y)_+)(y) .$$

Hence, the corrected approximation \widehat{Q} is defined as follows :

Definition 5 (Corrected approximation).

$$\widehat{Q}f(y) := Qf(y) + \begin{cases} E((p^y)_+)(y), & y \in \mathcal{G} \\ 0, & \text{otherwise} \end{cases} .$$

Thus we get

Theorem 2 (Corrected approximation errors). *Let $y \mapsto p^y$ be a smooth M -th order approximation to r . Then there exists a constant $C_3 > 0$ such that*

$$|\widehat{E}f(y)| = |f(y) - \widehat{Q}f(y)| \leq C_3 \cdot (\|g\|_{C^{M+1}} + \|r\|_{C^{M+1}}) \cdot h^{M+1} .$$

Definition 6 (The function \widehat{r}). *Given a smooth M -th order approximation to r , $y \mapsto p^y$, define $\widehat{r} : \mathcal{G} \rightarrow \mathbb{R}$ by*

$$\widehat{r}(y) = p^y(y) .$$

Theorem 3 (Smoothness of the approximation). *Assume that*

$$\nabla r(z) \neq 0 \quad , \quad \forall z \in \Gamma_r .$$

For small enough h , the corrected approximation term $\widehat{Q}f$ is a smooth function on

$$\mathbb{R}^n \setminus \{\widehat{r} = 0\} .$$

Theorem 4 (Approximation of the singularity location Γ_r). *Assume that*

$$\nabla r(z) \neq 0 \quad , \quad \forall z \in \Gamma_r .$$

Denote

$$\begin{cases} \mathcal{C}_1 := \{y \in \mathcal{G} : r(y) = 0\} \\ \mathcal{C}_2 := \{y \in \mathcal{G} : \widehat{r}(y) = 0\} \end{cases} ,$$

then

$$d_H(\mathcal{C}_1, \mathcal{C}_2) = O(h^{M+1}),$$

where d_H is the Hausdorff distance of the two sets.

Remark 3. *Although the results in this section were proven for quasi-interpolations with basis functions of finite support, they may also be proven for quasi-interpolations with basis functions of exponential decay. In the Numerical results (Section 7), we have used MLS quasi-interpolation with weight function of exponential decay.*

3 Proofs

3.1 Proof of Theorem 1

Proof 1. *Pick*

$$y \in \mathcal{G}((R+1)h) = \bigcup_{z \in \Gamma_r} B_{(R+1)h}(z) .$$

Then, there exists $z \in \Gamma_r$ with $\|y-z\| < (R+1)h$. By our assumption $\nabla r(z) \neq 0$. Using Taylor's approximation we can see that for any point

$$u \in B_{\Upsilon_M h} \left(z + (\Upsilon_M + 1)h \cdot \frac{\nabla r(z)}{\|\nabla r(z)\|} \right)$$

we have

$$r(u) = r(z) + \langle \nabla r(z), u - z \rangle + O(h^2) \geq h \|\nabla r(z)\| + O(h^2) .$$

Hence,

$$B_{\Upsilon_M h} \left(z + (\Upsilon_M + 1)h \cdot \frac{\nabla r(z)}{\|\nabla r(z)\|} \right) \cap X \subset \mathcal{P} .$$

Similarly we may show that

$$B_{\Upsilon_M h} \left(z - (\Upsilon_M + 1)h \cdot \frac{\nabla r(z)}{\|\nabla r(z)\|} \right) \cap X \subset X \setminus \mathcal{P} .$$

The intersections of each of the balls $B_{\Upsilon_M h} \left(z \pm (\Upsilon_M + 1)h \cdot \frac{\nabla r(z)}{\|\nabla r(z)\|} \right)$ with X must be uni-solvent for $\Pi_M(\mathbb{R}^n)$, thus both

$$B_{(R+2\Upsilon_M+2)h}(y) \cap \mathcal{P} \quad \text{and} \quad B_{(R+2\Upsilon_M+2)h}(y) \cap (X \setminus \mathcal{P})$$

are uni-solvent for $\Pi_M(\mathbb{R}^n)$ and $y \in \mathcal{G}$.

3.2 Proof of Theorem 2

Proof 2. *If $y \in \mathbb{R}^n \setminus \mathcal{G}$ then*

$$B_{(R+1)h}(y) \cap \Gamma_r = \emptyset ,$$

thus the restriction of f to $B_{Rh}(y)$ is smooth and

$$\left| \widehat{E}f(y) \right| = |Ef(y)| \leq C_1 \cdot \|f\|_{C^{M+1}} \cdot h^{M+1} \leq C_1 \cdot (\|g\|_{C^{M+1}} + \|r\|_{C^{M+1}}) \cdot h^{M+1} .$$

Otherwise, if $y \in \mathcal{G}$, then by our assumptions,

$$|p^y(u) - r(u)| \leq C_2 \cdot (\|g\|_{C^{M+1}} + \|r\|_{C^{M+1}}) \cdot h^{M+1} , \quad \forall u \in B_{Rh}(y) .$$

Consequently, we get,

$$|E(r_+)(y) - E(p^y_+)(y)| \leq (1 + L_1) \cdot C_2 \cdot (\|g\|_{C^{M+1}} + \|r\|_{C^{M+1}}) \cdot h^{M+1} .$$

Therefore,

$$\begin{aligned}
|\widehat{E}f(y)| &= |f(y) - \widehat{Q}f(y)| \\
&= |\underbrace{f(y) - Qf(y)}_{Ef(y)} - E((p^y)_+)(y)| \\
&= |Ef(y) - E((p^y)_+)(y)| \\
&\leq |Er_+(y) - E((p^y)_+)(y)| + |Eg(y)| \\
&\leq (1 + L_1) \cdot C_2 \cdot (\|g\|_{C^{M+1}} + \|r\|_{C^{M+1}}) \cdot h^{M+1} + C_1 \|g\|_{C^{M+1}} h^{M+1} .
\end{aligned}$$

□

3.3 Proof of Theorem 3

Proof 3. For $y \in \text{int}(\mathbb{R}^n \setminus \mathcal{G})$, there exists $\delta > 0$ such that within the ball

$$B_\delta(y) \subset \mathbb{R}^n \setminus \mathcal{G}$$

the correction term is equal to

$$\widehat{Q}f = Qf ,$$

which is a smooth function.

Similarly, for $y \in \text{int}(\mathcal{G}) \setminus \{\widehat{r} = 0\}$, there exists $\delta > 0$ such that

$$\forall v \in B_\delta(y) \subset \mathcal{G} \setminus \{\widehat{r} = 0\} ,$$

the correction term is equal to

$$\widehat{Q}f(v) = Qf(v) + E((p^v)_+)(v) = Qf(v) + (p^v)_+(v) - Q((p^v)_+)(v) .$$

The function Qf is obviously smooth. Likewise, the smoothness of $Q((p^v)_+)(v)$ follows from the smoothness of the mapping $v \mapsto p^v$. Last, since

$$B_\delta(y) \cap \{\widehat{r} = 0\} = \emptyset ,$$

then $(p^v)_+(v)$ is also smooth.

We still have to show that the corrected term is smooth for $y \in \partial(\mathcal{G})$. Since $\mathcal{G}((R+1)h) \subset \mathcal{G}$, we must have

$$B_{(R+1)h}(y) \cap \Gamma_r = \emptyset .$$

Specifically, for $v \in B_{\frac{1}{2}h}(y)$ we get

$$B_{(R+\frac{1}{2})h}(v) \cap \Gamma_r = \emptyset .$$

By our assumptions,

$$\nabla r(z) \neq 0 , \quad \forall z \in \Gamma_r ,$$

hence, since $y \mapsto p^y$ is a smooth M -th order approximation of r , we get for small enough h that

$$B_{Rh}(v) \cap \{p^v = 0\} = \emptyset .$$

That is, p^v does not change sign in $B_{Rh}(v)$, and $E((p^v)_+)(v) = 0$, which gives us

$$\widehat{Q}f(v) = Qf(v) , \quad \forall v \in B_{\frac{1}{2}h}(y) .$$

Thus the correction term is smooth at y .

□

3.4 Proof of Theorem 4

Proof 4. Pick $y \in \mathcal{C}_1$, then $r(y) = 0$ and $\nabla r(y) \neq 0$. Using Taylor's approximation we have

$$r(y \pm \epsilon \nabla r(y)) = \pm \epsilon \|\nabla r(y)\|^2 + O(\epsilon^2 \cdot \|\nabla r(y)\|^2) . \quad (4)$$

Hence,

$$r(y - \epsilon \nabla r(y)) < 0 < r(y + \epsilon \nabla r(y)) .$$

Since $y \mapsto p^y$ is a smooth M -th order approximation to r , the function $\hat{r}(y) = p^y(y)$ must be smooth and satisfy

$$\hat{r}(y) - r(y) = O(h^{M+1}) . \quad (5)$$

hence

$$\hat{r}(y - \epsilon \nabla r(y)) < 0 < \hat{r}(y + \epsilon \nabla r(y)) .$$

Then there must exist $u = y + \lambda \nabla r(y)$ with $|\lambda| < \epsilon$ such that $\hat{r}(u) = 0$. However, from (4) and (5) we get

$$O(h^{M+1}) = r(u) = \lambda \|\nabla r(y)\|^2 + O(\lambda^2 \|\nabla r(y)\|^2) .$$

Hence,

$$\lambda \|\nabla r(y)\|^2 = O(h^{M+1}) ,$$

and consequently

$$\|y - u\| = |\lambda| \|\nabla r(y)\| = O(h^{M+1}) .$$

That is, there must exist u with $\|y - u\| = O(h^{M+1})$ and $\hat{r}(u) = 0$, hence,

$$\sup_{y \in \mathcal{C}_1} \inf_{u \in \mathcal{C}_2} d(y, u) = O(h^{M+1}) .$$

Similarly we show that

$$\sup_{y \in \mathcal{C}_2} \inf_{u \in \mathcal{C}_1} d(y, u) = O(h^{M+1}) ,$$

and we have

$$d_H(\mathcal{C}_1, \mathcal{C}_2) = O(h^{M+1}) .$$

□

4 Partitioning the data points in X with respect to the sign of r

Our method relies upon our ability to correctly identify a set $\mathcal{P} \subset X$ which partitions the data points in X with respect to the signs of the function r . That is,

1. For all $x \in \mathcal{P}$ either $r(x) > 0$ or $r(x) = O(h^{M+1})$.
2. For all $x \in X \setminus \mathcal{P}$ either $r(x) < 0$ or $r(x) = O(h^{M+1})$.

We propose an algorithm for building the set \mathcal{P} according to the following steps:

Step (1) Find a set $\mathcal{S} \subset X$ satisfying $\Gamma_r \subset \bigcup_{x \in \mathcal{S}} B_{3h}(x)$.

Step (2) Denote by A_1, \dots, A_k the connected components of

$$\mathbb{R}^n \setminus \left(\bigcup_{x \in \mathcal{S}} B_{3h}(x) \right) \subset \mathbb{R}^n \setminus \Gamma_r .$$

Note that for all $1 \leq i \leq k$ and $x_1, x_2 \in A_i$ we have $r(x_1) \cdot r(x_2) > 0$.

Step (3) Define a function $a : X \rightarrow \{1, \dots, k\}$ satisfying

$$\forall x \in X : a(x) = i \implies \begin{array}{ll} \text{either} & r(x) = O(h^{M+1}) \\ \text{or} & r(x) \cdot r(u) > 0, \forall u \in A_i \cap X \end{array} .$$

For $1 \leq i \leq k$ set $\bar{A}_i := \{x \in X : a(x) = i\}$.

Step (4) Define a function $\sigma : \{1, \dots, k\} \rightarrow \{1, 2\}$ satisfying

$$\sigma(i_1) = \sigma(i_2) \implies \forall x_1 \in A_{i_1}, \forall x_2 \in A_{i_2} : r(x_1) \cdot r(x_2) > 0 .$$

Step (5) Set $\mathcal{P} := \bigcup_{\sigma(i)=1} \bar{A}_i$.

Refine the set \mathcal{P} .

Step 1 Build the set \mathcal{S} satisfying $\Gamma_r \subset \bigcup_{x \in \mathcal{S}} B_{3h}(x)$

1: Define $\Psi : \mathbb{R}^n \rightarrow \Pi_M(\mathbb{R}^n)$ by

$$\Psi(y) := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in X} \omega\left(\frac{\|x - y\|}{h}\right) \cdot (p(x) - f(x))^2 .$$

Here ω is a smooth weight function with compact support

$$\text{supp}(\omega) = [0, \rho] \supset [0, \Upsilon_M] .$$

2: For $u \in X$ define $\epsilon_u := \max \{ |(\Psi(u))(x) - f(x)| : x \in B_{\rho h}(u) \cap X \}$.

3: Set

$$\mathcal{S} := \left\{ x \in X \left| \exists u \in B_{(\rho+5)h}(x) \cap X : |(\Psi(u))(x) - f(x)| > \epsilon_u \cdot \left(\frac{\rho+5}{\rho} \right)^{M+1} \right. \right\} .$$

Step 2 Find the connected components of $\mathbb{R}^n \setminus \left(\bigcup_{x \in \mathcal{S}} B_{3h}(x) \right)$

1: Build a graph $G = (V, E)$ with vertices

$$V := X \setminus \left(\bigcup_{x \in \mathcal{S}} B_{3h}(x) \right) ,$$

and edges

$$E := \{(u, v) \in V \mid d(u, v) < 2h\} .$$

2: Set A_1, A_2, \dots, A_k to be the connected components of the graph G .

Step 3 Compute the function $a : X \rightarrow \{1, \dots, k\}$ satisfying :
 $a(x) = i \Rightarrow$ either $r(x) = O(h^{M+1})$ or $r(x) \cdot r(u) > 0$, $\forall u \in A_i \cap X$.

1: Let ω be a smooth weight function with compact support.
 2: For $1 \leq i \leq k$ denote

$$\Omega_i = \left\{ y \in \mathbb{R}^n : \text{supp} \left(\omega \left(\frac{\|\cdot - y\|}{h} \right) \right) \cap A_i \cap X \text{ is uni-solvent for } \Pi_M(\mathbb{R}^n) \right\} ,$$

and define $\Theta_i : \Omega_i \rightarrow \Pi_M(\mathbb{R}^n)$ by

$$\Theta_i(y) := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in A_i \cap X} \omega \left(\frac{\|x - y\|}{h} \right) \cdot (p(x) - f(x))^2 .$$

3: For $x \in X$ define

$$a(x) := \begin{cases} i & , x \in A_i \\ \arg \min_{\{1 \leq i \leq k : x \in \Omega_i\}} |(\Theta_i(x))(x) - f(x)| & , x \in X \setminus \left(\bigcup_{i=1}^k A_i \right) \end{cases} .$$

Step 4 Compute the function $\sigma : \{1, \dots, k\} \rightarrow \{1, 2\}$ satisfying :
 $\sigma(i_1) = \sigma(i_2) \Rightarrow \forall x_1 \in A_{i_1}, \forall x_2 \in A_{i_2} : r(x_1) \cdot r(x_2) > 0$.

- 1: Define $\sigma^0 : \{1, \dots, k\} \rightarrow \{1, \dots, k\}$ by $\sigma^0(i) = i$.
- 2: For $1 \leq l \leq k$ define $X_l^0 = A_l$.
- 3: **for** $j = 0, \dots, k-3$ **do**
- 4: For each $1 \leq l \leq k-j$ denote $X_l^{j+1} := \bigcup_{\sigma^j(i)=l} X_i^j$.
- 5: For each $1 \leq l \leq k-j$ define $\Phi_l : \mathbb{R}^n \rightarrow \Pi_M(\mathbb{R}^n)$ by

$$\Phi_l(y) := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in X_l^{j+1}} \omega \left(\frac{\|x - y\|}{h} \right) \cdot (p(x) - f(x))^2.$$

Here ω is a smooth weight function with compact support $\text{supp}(\omega) = [0, \rho]$.

- 6: **for** $1 \leq l_1, l_2 \leq k-j$ **do**
- 7: $N \leftarrow \left\{ x \in X_{l_1}^{j+1} : B_{\rho h}(x) \cap X_{l_2}^{j+1} \neq \emptyset \right\}$
- 8: **if** $\#N == 0$ **then**
- 9: $D_{l_1, l_2} \leftarrow \infty$
- 10: **else**
- 11: $D_{l_1, l_2} \leftarrow \max \{ |(\Phi_{l_2}(x))(x) - f(x)| : x \in N \}$
- 12: **end if**
- 13: **end for**
- 14: Pick $1 \leq l_1 < l_2 \leq k-j$ for which $(D + D^T)_{l_1, l_2}$ is minimal.
- 15: Define $\sigma^{j+1} : \{1, \dots, k-j\} \rightarrow \{1, \dots, k-j-1\}$ by

$$\sigma^{j+1}(i) = \begin{cases} i, & 1 \leq i \leq l_2 - 1 \\ l_1, & i = l_2 \\ i - 1, & l_2 < i \leq k-j \end{cases}.$$

- 16: **end for**
 - 17: Set $\sigma = \sigma^{k-2} \circ \sigma^{k-3} \circ \dots \circ \sigma^0$.
-

Step 5 Refinement of the set \mathcal{P}

- 1: $B \leftarrow \{x \in X : \exists x_1 \in \mathcal{P}, x_2 \in X \setminus \mathcal{P} \text{ s.t. } d(x, x_1), d(x, x_2) < 2h\}$
- 2: $O_1 \leftarrow \mathcal{P} \setminus B$
- 3: $O_2 \leftarrow X \setminus (\mathcal{P} \cup B)$
- 4: For $k = 1, 2$ define $\Xi_k : \mathbb{R}^n \rightarrow \Pi_M(\mathbb{R}^n)$ by

$$\Xi_k(y) := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in O_k} \omega \left(\frac{\|x - y\|}{h} \right) \cdot (p(x) - f(x))^2.$$

Here ω is a smooth weight function with compact support.

- 5: $\mathcal{P} \leftarrow O_1 \cup \{x \in B : |(\Xi_1(x))(x) - f(x)| < |(\Xi_2(x))(x) - f(x)|\}$.
-

To show that the algorithm indeed generates the set \mathcal{P} , as defined in Definition 2, we observe the following:

1. In Step 1, Ψ is an MLS polynomial approximation of f , hence if

$$B_{\rho h}(u) \cap \Gamma_r = \emptyset$$

then the restriction of f to $B_{\rho h}(u)$ is either g or $g + r$. W.l.o.g. we will assume that

$$r(y) < 0, \quad \forall y \in B_{\rho h}(u),$$

hence the restriction of f to $B_{\rho h}(u)$ is the smooth function g and

$$|(\Psi(u))(y) - g(y)| = O(h^{M+1}), \quad \forall y \in B_{(\rho+5)h}(u) \quad (6)$$

Specifically, for $x \in B_{\rho h}(u) \cap X$ we have $f(x) = g(x)$, and

$$\epsilon_u = \max \{ |(\Psi(u))(x) - f(x)| : x \in B_{\rho h}(u) \cap X \} = O(h^{M+1}) \quad (7)$$

However, if

$$\exists z \in B_{(\rho+2)h}(u) \cap \Gamma_r \quad \text{with} \quad \nabla r(z) \neq 0$$

then there must exist

$$x \in B_h \left(z + 2h \cdot \frac{\nabla r(z)}{\|\nabla r(z)\|} \right) \cap X \subset B_{(\rho+5)h}(u) \cap X$$

for which

$$r(x) > h \cdot \|\nabla r(z)\| + O(h^2) > 0.$$

From (6) we get

$$\begin{aligned} |(\Psi(u))(x) - f(x)| &= |(\Psi(u))(x) - g(x) - r(x)| \\ &> |r(x)| - |(\Psi(u))(x) - g(x)| \\ &= h \cdot \|\nabla r(z)\| + O(h^2) + O(h^{M+1}) \end{aligned}$$

For small enough h we get by (7)

$$|(\Psi(u))(x) - f(x)| > h \cdot \|\nabla r(z)\| + O(h^2) > \epsilon_u \cdot \left(\frac{\rho+5}{\rho} \right)^{M+1}.$$

Thus $x \in \mathcal{S}$ and $z \in \bigcup_{x \in \mathcal{S}} B_{3h}(x)$.

2. In Step 3, each Θ_i is an MLS approximation of f based only upon the data points in $A_i \cap X$. Since $A_i \cap \Gamma_r = \emptyset$, f is smooth on each A_i and Θ_i is either a polynomial approximation of g or of $g + r$.

Pick $x \in \mathcal{S}$ and assume that both x and A_i belong to the same connected component of $\mathbb{R}^n \setminus \Gamma_r$, hence

$$|(\Theta_i(x))(x) - f(x)| = O(h^{M+1}).$$

W.l.o.g. assume that $r(x) > 0$. If $r(u) < 0$ for $u \in A_{a(x)}$ then $\Theta_{a(x)}(x)$ is a polynomial approximation of g , which gives us

$$\begin{aligned} O(h^{M+1}) &= |(\Theta_i(x))(x) - f(x)| \\ &\geq |(\Theta_{a(x)}(x))(x) - f(x)| \\ &\geq |r(x)| - |(\Theta_{a(x)}(x))(x) - g(x)| = |r(x)| + O(h^{M+1}) \end{aligned}$$

Thus in this case $r(x) = O(h^{M+1})$.

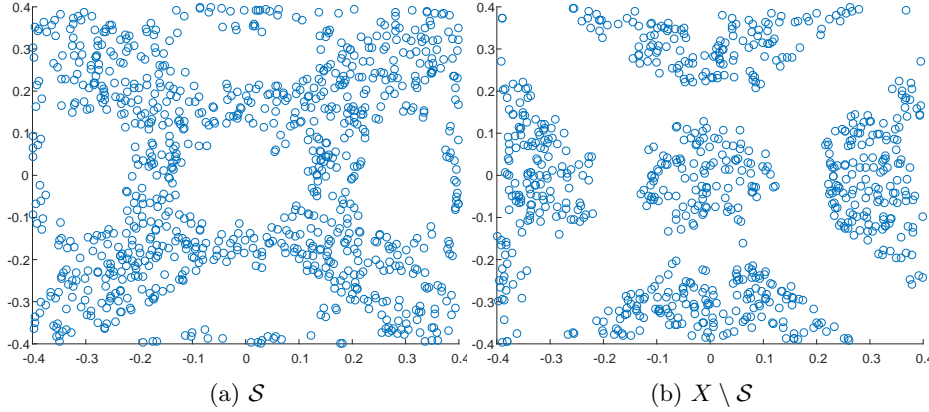


Figure 1: The sets \mathcal{S} and $X \setminus \mathcal{S}$.

3. In Step 4, the operator Φ_l returns MLS approximations of either g or $g + r$. At each step of the for-loop we merge two subsets $X_{l_1}^{j+1}$ and $X_{l_2}^{j+1}$ on which the approximations Φ_{l_1} and Φ_{l_2} are close. At the end we will have two subsets of X , such the the restriction of f to one subset would be g and the restriction of f to the other subset would be $g + r$.
4. In Step 5 we propose to refine the initial set \mathcal{P} . We refine this set by removing from \mathcal{P} and from $X \setminus \mathcal{P}$ data points that are close to the boundary. Then, we add to \mathcal{P} only the boundary data points for which the MLS approximation based upon \mathcal{P} has smaller errors than the MLS approximation based upon $X \setminus \mathcal{P}$.

The above observations prove that the set \mathcal{P} returned by our proposed algorithm is a partition of the data points in X with respect for the sign of the function r .

Remark 4. In Step Step (5) of the algorithm we arbitrarily choose the set \mathcal{P} , thus we might choose the set on which r returns negative values. However, this choice has no effect on the approximation algorithm. Indeed,

$$Er_+ = E(r - r_-) = \underbrace{Er}_{O(h^{M+1})} + \underbrace{E(-(r_-))}_{(-r)_+} = O(h^{M+1}) + E(-r)_+,$$

where E is the approximation error of the quasi-interpolation operator. Hence, the initial choice is insignificant.

For example, we ran the partitioning algorithm on the function

$$f(x, y) = ((x + y) \cdot (x - y))_+.$$

In Figure 1 one can see the sets \mathcal{S} and $X \setminus \mathcal{S}$ (see Step 1). In Figure 2 one can see the connected components of $X \setminus \mathcal{S}$ (Step 2), and the connected components of $X \setminus \Gamma_r$ (see Step 3). In Figure 3 one can see the initial set \mathcal{P} and the final \mathcal{P} after the refinement (see Step 5).

Remark 5. Although the sign determination algorithm, is described for f defined on \mathbb{R}^n , the algorithm can also be applied to a compact domain. Moreover,

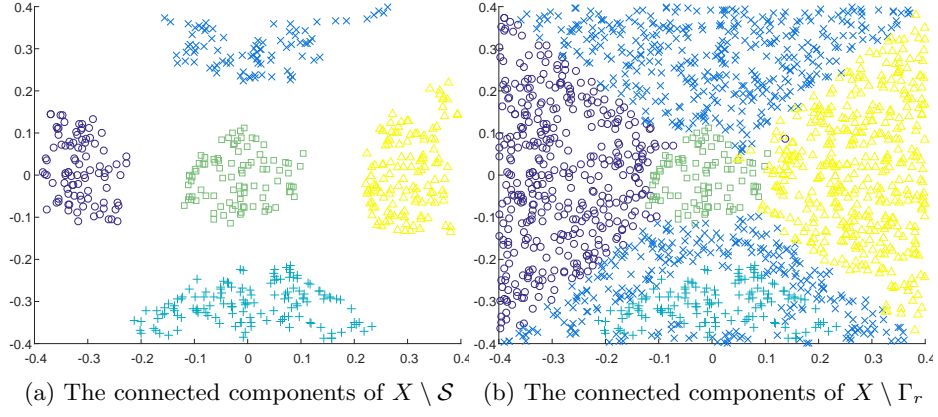


Figure 2: The connected components of $X \setminus S$ and of $X \setminus \Gamma_r$.

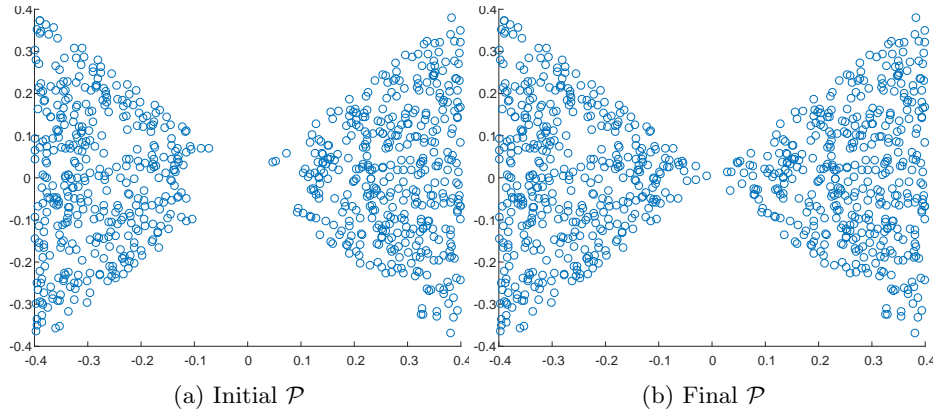


Figure 3: The set \mathcal{P} before and after the refinement.

since the computation of the corrected approximation $\hat{Q}f$, is a local procedure, it might be computationally preferable to break the domain into smaller compact subsets.

5 Approximation of the signed function r

In Section 2 we have introduced the notion of a smooth M -th order approximation to the function r . In this section we will introduce two methods we may use to find a smooth M -th order approximation to r . Recall that a smooth M -th order approximation is a mapping $y \mapsto p^y$ satisfying :

- There exists $C_2 > 0$ such that for all $y \in \mathcal{G}$,

$$|p^y(u) - r(u)| \leq C_2 \cdot (\|g\|_{C^{M+1}} + \|r\|_{C^{M+1}}) \cdot h^{M+1}, \quad \forall u \in B_{Rh}(y).$$

- Let $\{p_\alpha\}_{\alpha \in I}$ be a polynomial basis of $\Pi_M(\mathbb{R}^n)$ and write

$$p^y = \sum_{\alpha \in I} \lambda_\alpha(y) \cdot p_\alpha.$$

Then, the mappings $y \mapsto \lambda_\alpha(y)$ are infinitely smooth.

The methods that we introduce are both based upon MLS, hence the smoothness is trivial. As for the first condition, while the first method gives slightly better approximations to r (as can be seen in section 7), it also depends upon a non-singularity conjecture (which can be verified numerically). The first method, described in subsection 5.1, is in-fact a generalization of the method suggested by Lipman and Levin [10] for the univariate case. Here also we analyse the quasi-interpolation errors, and find a polynomial p_1^y which gives the best simulation for these errors. The second method, described in subsection 5.2 utilizes two MLS approximations and is independent upon a non-singularity condition. One MLS approximation is based upon the data set \mathcal{P} and the other is based upon its complement $X \setminus \mathcal{P}$. The difference between the two approximations would be the polynomial approximation to r .

5.1 Approximation by error analysis

This method derives the polynomial approximation to r from an analysis of the quasi-interpolation errors.

Definition 7 (The approximant p_1^y). For $y \in \mathbb{R}^n$ define $p_1^y \in \Pi_M(\mathbb{R}^n)$ by

$$p_1^y := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in X} \omega \left(\frac{\|y - x\|}{h} \right) \cdot (E(\chi_{\mathcal{P}} \cdot p)(x) - Ef(x))^2.$$

Here $\omega : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is an infinitely smooth positive weight function with compact support,

$$\text{supp}(\omega) \supset [0, (R + 2\Upsilon_M + 2)h]$$

and $\chi_{\mathcal{P}}$ is the indicator function defined by

$$\chi_{\mathcal{P}}(z) = \begin{cases} 1, & z \in \mathcal{P} \\ 0, & z \notin \mathcal{P} \end{cases}, \quad \forall z \in \mathbb{R}^n.$$

Remark 6. A more natural choice for the polynomial p_1^y would have been the polynomial minimizing the sum

$$\sum_{x \in X} \omega \left(\frac{\|y - x\|}{h} \right) \cdot (E(p_+)(x) - Ef(x))^2 .$$

However, this computation is not linear.

Theorem 5 (p_1^y is a smooth M -th order approximation to r). If the vectors

$$\{E(\chi_{\mathcal{P}} \cdot p_{\alpha})(x) : x \in \Omega(y) \cap X\}_{\alpha \in I}$$

are linearly independent for all $y \in \mathcal{G}$, with

$$\Omega(y) = \left\{ y + z : \frac{\|z\|}{h} \in \text{supp}(\omega) \right\} ,$$

then the mapping $y \mapsto p_1^y$ is a smooth M -th order approximation to r .

Proof 5. Let us define an MLS operator \tilde{Q} for a function $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ and $y \in \mathbb{R}^n$ by

$$\tilde{Q}\phi(y) = p_{\phi,y}(y)$$

where

$$p_{\phi,y} := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in X} \omega \left(\frac{\|y - x\|}{h} \right) \cdot (E(\chi_{\mathcal{P}} \cdot p)(x) - E(\chi_{\mathcal{P}} \cdot \phi)(x))^2 .$$

Note the difference between the above definition and the original MLS definition (3). In the original MLS definition we sum the squares of the difference between the values of ϕ and of the approximating polynomial p at the data points, while in this definition we sum the squares of the differences between the approximation errors of $\chi_{\mathcal{P}} \cdot \phi$ and $\chi_{\mathcal{P}} \cdot p$ at the data points. Write

$$p = \sum_{\alpha \in I} \delta_{\alpha} p_{\alpha} \quad \text{and} \quad p_{\phi,y} = \sum_{\alpha \in I} (\delta_{\phi,y})_{\alpha} p_{\alpha} ,$$

then from the linearity of E we have

$$\delta_{\phi,y} = \arg \min_{\delta} \sum_{x \in X} \omega \left(\frac{\|y - x\|}{h} \right) \cdot \left(\sum_{\alpha} \delta_{\alpha} E(\chi_{\mathcal{P}} \cdot p_{\alpha})(x) - E(\chi_{\mathcal{P}} \cdot \phi)(x) \right)^2 .$$

To solve this problem we follow [9], from which we know that if the vectors

$$\{E(\chi_{\mathcal{P}} \cdot p_{\alpha})(x) : x \in \Omega(y) \cap X\}_{\alpha \in I}$$

are linearly independent then the solution to this problem is the vector $\delta_{\phi,y}$ defined by

$$\delta_{\phi,y} = (ADA^T)^{-1} AD\phi \tag{8}$$

where A is the matrix

$$A = (E(\chi_{\mathcal{P}} \cdot p_{\alpha})(x))_{\alpha,x} , \quad \forall \alpha \in I , \quad \forall x \in X$$

D is the diagonal matrix with values

$$D_{x,x} = \omega \left(\frac{\|y - x\|}{h} \right), \forall x \in X$$

and ϕ is the vector

$$\phi_x = E(\chi_{\mathcal{P}} \cdot \phi)(x), \forall x \in X.$$

Moreover, the operator \tilde{Q} clearly reproduces polynomials in $\Pi_M(\mathbb{R}^n)$ and as such is a quasi-interpolation operator

$$\tilde{Q}\phi(y) = \sum_{x \in X} \tilde{q}_x(y) E(\chi_{\mathcal{P}} \cdot \phi)(x) \quad (9)$$

with basis functions

$$\tilde{q}_x(y) := \sum_{\alpha \in I} p_{\alpha}(y) \cdot ((ADA^T)^{-1}AD)_{\alpha,x} \quad (10)$$

of compact support

$$R_2 := \inf \{ \rho : \text{supp}(\tilde{q}_x) \subseteq B_{\rho h}(x), \forall x \in X \}.$$

Note that

$$p_1^y(z) = \sum_{x \in X} \tilde{q}_x(z) \cdot E f(x). \quad (11)$$

Let us show that \tilde{Q} has a bounded Lebesgue constant,

$$L_2 := \sup \left\{ \sum_{x \in X} |\tilde{q}_x(z)| : z \in \mathbb{R}^n \right\}.$$

Assume that the polynomials $\{p_{\alpha}\}$ are each of the form

$$p_{\alpha}(u) = (u - z)^{\alpha},$$

with α a multi-index. Then we may write

$$A = H \cdot F$$

where H is the diagonal matrix with values

$$H_{\alpha,\alpha} = h^{|\alpha|}, \forall \alpha \in I$$

and F is the matrix

$$F = \left(E \left(\chi_{\mathcal{P}} \cdot \left(\frac{\cdot - z}{h} \right)^{\alpha} \right) (x) \right)_{\alpha,x}, \forall \alpha \in I, \forall x \in X.$$

From (1) and (2) we get

$$|F_{\alpha,x}| = \left| E \left(\chi_{\mathcal{P}} \cdot \left(\frac{\cdot - z}{h} \right)^{\alpha} \right) (x) \right| \leq (1 + L_1) \cdot R^M.$$

Then,

$$\begin{aligned}
|\tilde{q}_x(z)| &:= \left| \sum_{\alpha \in I} p_\alpha(z) \cdot ((ADA^T)^{-1}AD)_{\alpha,x} \right| = \\
&\left| \sum_{\alpha \in I} p_\alpha(z) \cdot (((HF)D(HF)^T)^{-1}(HF)D)_{\alpha,x} \right| = \\
&\left| \sum_{\alpha \in I} p_\alpha(z) \cdot (H^{-1}(FDF^T)^{-1}H^{-1}HFD)_{\alpha,x} \right| = \\
&\left| \sum_{\alpha \in I} p_\alpha(z) \cdot (H^{-1}(FDF^T)^{-1}FD)_{\alpha,x} \right| = \\
&\left| \sum_{\alpha \in I} \left(\frac{z-x}{h} \right)^\alpha \cdot ((FDF^T)^{-1}FD)_{\alpha,x} \right| \leq \\
&\|(FDF^T)^{-1}\| \cdot (1 + L_1) \cdot R^M \cdot \omega\left(\frac{|z-x|}{h}\right)
\end{aligned}$$

with

$$\|(FDF^T)^{-1}\| = \sup \left\{ \frac{\|(FDF^T)^{-1}\mathbf{v}\|}{\|\mathbf{v}\|} : \mathbf{v} \neq \mathbf{0} \right\},$$

Note that while the term $\|(FDF^T)^{-1}\|$ is independent upon the value of h , it is dependent upon the distribution of the data points. Thus,

$$L_2 \leq \|(FDF^T)^{-1}\| \cdot (1 + L_1) \cdot R^M \cdot K,$$

where

$$K := \sup \left\{ \sum_{x \in X} \omega\left(\frac{z-x}{h}\right) : z \in \mathbb{R}^n \right\}.$$

That is, the operator \tilde{Q} has a bounded Lebesgue constant. Note that this constant is independent of the choice of the polynomial basis $\{p_\alpha\}_{\alpha \in I}$.

From (11) we have

$$p_1^y(z) = \sum_{x \in X} \tilde{q}_x(z) \cdot Ef(x).$$

Also, since the operator \tilde{Q} reproduces polynomials we have that

$$p_y^{Taylor}(z) = \sum_{x \in X} \tilde{q}_x(z) \cdot E(\chi_{\mathcal{P}} \cdot p_y^{Taylor})(x),$$

where p_y^{Taylor} is the Taylor approximation of the function r at the point y .

Pick $z \in B_{Rh}(y)$, then

$$\begin{aligned}
|p_1^y(z) - p_y^{Taylor}(z)| &= \\
&\left| \sum_{x \in X} \tilde{q}_x(z) \cdot (Ef(x) - E(\chi_{\mathcal{P}} \cdot p_y^{Taylor})(x)) \right| \leq \\
&L_2 \cdot \max_{x \in X \cap B_{R_2h}(z)} \{|Ef(x) - E(\chi_{\mathcal{P}} \cdot p_y^{Taylor})(x)|\}
\end{aligned}$$

Compute,

$$\begin{aligned}
& |Ef(x) - E(\chi_{\mathcal{P}} \cdot p_y^{Taylor})(x)| = \\
& |Eg(x) + Er_+(x) - E(\chi_{\mathcal{P}} \cdot p_y^{Taylor})(x)| \leq \\
& |Eg(x)| + |Er_+(x) - E(\chi_{\mathcal{P}} \cdot p_y^{Taylor})(x)| = \\
& |Eg(x)| + |E(\chi_{\mathcal{P}} \cdot r)(x) - E(\chi_{\mathcal{P}} \cdot p_y^{Taylor})(x)| = \\
& |Eg(x)| + |E(\chi_{\mathcal{P}} \cdot (r - p_y^{Taylor}))(x)| \leq \\
& C_1 \|g\|_{C^{M+1}} h^{M+1} + (1 + L_1) \|r\|_{C^{M+1}} \|x - y\|^{M+1}
\end{aligned}$$

Hence for all $z \in B_{Rh}(y)$ we have

$$\begin{aligned}
& |p_1^y(z) - r(z)| \leq \\
& |p_1^y(z) - p_y^{Taylor}(z)| + |p_y^{Taylor}(z) - r(z)| \leq \\
& L_2 \left(C_1 \|g\|_{C^{M+1}} + (1 + L_1) \|r\|_{C^{M+1}} (R + R_2)^{M+1} \right) h^{M+1} + \\
& R^{M+1} \|r\|_{C^{M+1}} h^{M+1}
\end{aligned}$$

Clearly for

$$C_2 = \max \{ L_2 C_1, \quad L_2 (1 + L_1) (R + R_2)^{M+1} + R^{M+1} \}$$

we get for all $z \in B_{Rh}(y)$ that

$$|p_1^y(z) - r(z)| \leq C_2 \cdot (\|g\|_{C^{M+1}} + \|r\|_{C^{M+1}}) \cdot h^{M+1}.$$

Moreover, from (11) and from (10) we get the representation

$$p_1^y = \sum_{\alpha \in I} p_{\alpha} \cdot \underbrace{\sum_{x \in X} ((ADA^T)^{-1} AD)_{\alpha, x} Ef(x)}_{\lambda_{\alpha}(y)},$$

from which it is clear that the mappings of the coefficients $y \mapsto \lambda_{\alpha}(y)$ are smooth. \square

Remark 7. In Theorem 5 we have assumed that the vectors

$$\{E(\chi_{\mathcal{P}} \cdot p_{\alpha})(x) : x \in \Omega(y) \cap X\}_{\alpha \in I}$$

are linearly independent. While it seems intuitive that a truncated polynomial can not be approximated by polynomials, we did not succeed in proving this. Thus we leave it as a conjecture.

Conjecture 6 (Linear independence). For any point $y \in \mathcal{G}$, denote

$$\Omega(y) = \left\{ y + z : \frac{\|z\|}{h} \in \text{supp}(\omega) \right\}.$$

Then, the vectors

$$\{E(\chi_{\mathcal{P}} \cdot p_{\alpha})(x) : x \in \Omega(y) \cap X\}_{\alpha \in I}$$

are linearly independent.

We also suggest a second method for approximating r which does not depend upon the above conjecture.

5.2 Approximation by partitioned MLS

This method utilizes two MLS approximations, the first is based only upon the data points in \mathcal{P} , while the second is based only upon the data points in $X \setminus \mathcal{P}$.

Definition 8 (The approximant p_2^y). *For a point $y \in \mathcal{G}$ define $p_2^y := T_1(f) - T_2(f)$ where*

$$T_1(f) := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in \mathcal{P}} \omega \left(\frac{\|y - x\|}{h} \right) \cdot (p(x) - f(x))^2 ,$$

and

$$T_2(f) := \arg \min_{p \in \Pi_M(\mathbb{R}^n)} \sum_{x \in X \setminus \mathcal{P}} \omega \left(\frac{\|y - x\|}{h} \right) \cdot (p(x) - f(x))^2 .$$

Here ω is a weight function satisfying $\text{supp}(\omega) \supset [0, (R + 2\Upsilon_M + 2)h]$.

Theorem 7 (p_2^y is a smooth M -th order approximation to r). *The mapping $y \mapsto p_2^y$ is a smooth M -th order approximation to r .*

Proof 6. *Pick $y \in \mathcal{G}$, by Definition 3, both*

$$B_{(R+2\Upsilon_M+2)h}(y) \cap \mathcal{P} \quad \text{and} \quad B_{(R+2\Upsilon_M+2)h}(y) \cap (X \setminus \mathcal{P})$$

are uni-solvent for $\Pi_M(\mathbb{R}^n)$.

The operator $T_1 : \mathbb{R}^X \rightarrow \Pi_M(\mathbb{R}^n)$ is an MLS operator, and as such there exists a constant c_1 satisfying

$$|T_1(\phi)(u) - \phi(u)| \leq c_1 \cdot \|\phi\|_{C^{M+1}} \cdot h^{M+1} , \quad \forall u \in B_{Rh}(y) , \quad \phi \in C^{M+1}(\mathbb{R}^n) .$$

Note that the restriction of $f = g + r_+$ to \mathcal{P} satisfies

$$f(x) - g(x) - r(x) = O(h^{M+1}) , \quad \forall x \in \mathcal{P} .$$

Hence

$$T_1(f)(u) - T_1(g + r)(u) = O(h^{M+1}) , \quad \forall u \in B_{Rh}(y) .$$

Moreover, since $g, r \in C^{M+1}(\mathbb{R}^n)$,

$$T_1(f)(u) - g(u) - r(u) = O(h^{M+1}) , \quad \forall u \in B_{Rh}(y) .$$

Similarly we may show that

$$T_2(f)(u) - g(u) = O(h^{M+1}) , \quad \forall u \in B_{Rh}(y) .$$

Therefore,

$$\begin{aligned} |p_2^y(u) - r(u)| &= |T_1(f)(u) - T_2(f)(u) - r(u)| \\ &\leq |T_1(f)(u) - g(u) - r(u)| + |g(u) - T_2(f)(u)| = O(h^{M+1}) . \end{aligned}$$

Last note that since p_2^y is defined as the difference between two MLS solutions, the mappings from y to the coefficients of the polynomial p_2^y must be smooth.

6 Computational Complexity

The correction of the approximation at a point $y \in \mathbb{R}^n$ consists of three steps.

1. We partition the data points in $X \cap B_{(R+2\Upsilon_M+2)h}(y)$ with respect to the sign of the function r (see Section 4). This partitioning is based upon several MLS approximations, hence its complexity is $O(N \cdot (d^2 \cdot N + d^3 \cdot j))$ where

$$N := \max \{k \in \mathbb{N} : \forall y \in \mathbb{R}^n, \#B_{(R+2\Upsilon_M+2)h}(y) \cap X \leq k\} ,$$

j is the number of connected components of $B_{(R+2\Upsilon_M+2)h}(y) \setminus \Gamma_r$ and

$$d := \dim(\Pi_M(\mathbb{R}^n)) .$$

2. We find the approximating polynomial to the function r, p^y . In Section 5 we have proposed two methods for finding such a polynomial. Since both methods are essentially based upon MLS, they have the same complexity of $O(d^2 \cdot N + d^3)$.
3. Last, we set the corrected approximation to be

$$\widehat{Q}f(y) = Qf(y) + E((p^y)_+)(y) .$$

This step has a complexity of $O(N)$.

7 Numerical results

For our test we took our data set X to be a random set of 41^2 data points with fill distance 0.036 in the region

$$[-0.4, 0.4]^2 \subset \mathbb{R}^2 .$$

Using the sampled data points, we approximated a function, and compared our approximation to the actual function values. We performed this comparison on a uniform mesh of 81^2 points with fill distance $h = 0.01$ in the region

$$[-0.36, 0.36]^2 \subset \mathbb{R}^2 .$$

Note that we avoided testing points close to the edge, wishing to avoid approximation errors resulting from partial neighbourhood near the edges. For the quasi-interpolation we used the MLS approximation with weight function

$$\omega\left(\frac{\|y - x\|}{h}\right) = e^{-\frac{\|y - x\|^2}{40h^2}} .$$

We approximated the functions $f_k = g + (r_k)_+$ for $1 \leq k \leq 4$ where

$$g(x, y) = e^{x+y}$$

and

$$\bullet \quad r_1(x, y) = x^2 + y^2 - \frac{1}{5}^2$$

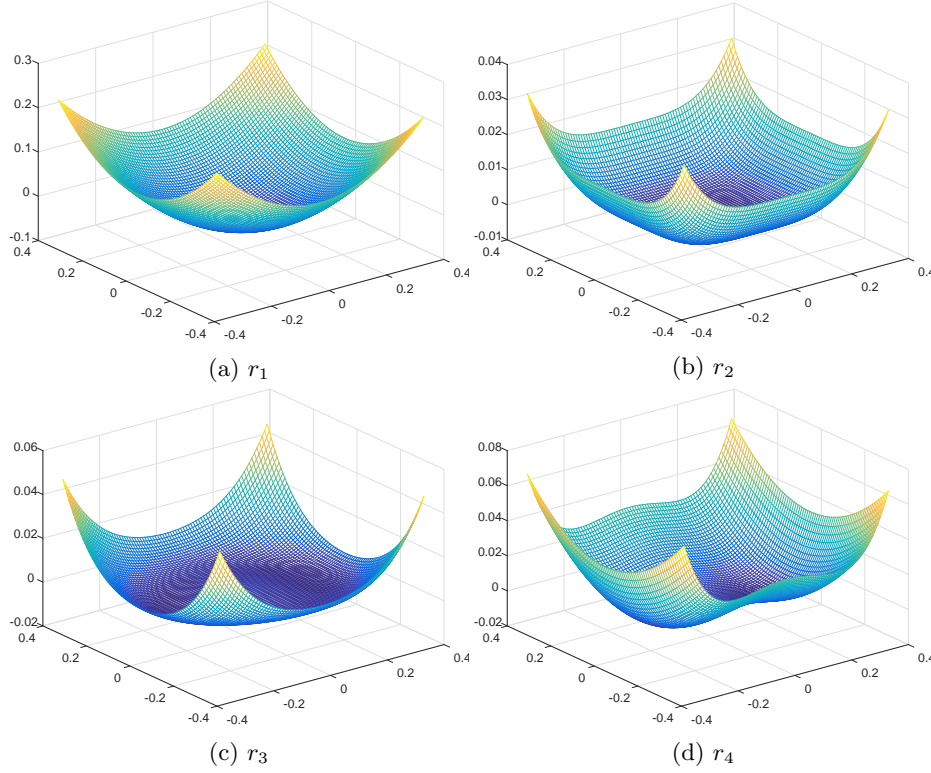


Figure 4: The functions r_k for $k = 1, \dots, 4$.

- $r_2(x, y) = x^4 + y^4 - \frac{1}{5}$
- $r_3(x, y) = \left(\left(x - \frac{1}{10} \right)^2 + y^2 - \frac{1}{5} \right) \cdot \left(\left(x + \frac{1}{10} \right)^2 + y^2 - \frac{1}{5} \right)$
- $r_4(x, y) = 4 \cdot x^4 + \frac{y^2 - x^2}{4}$

See fig. 4 for the graphs of these functions.

Unless otherwise specified, we have set the value of M , the maximal total degree of the approximating polynomials, to 4.

In figs. 5 to 8, one can see a comparison between the errors of the original MLS approximation, Ef , and our corrected approximations, $\hat{E}_1 f$ and $\hat{E}_2 f$, based upon p_1^y and p_2^y respectively.

Note that while the errors of the original approximation were distinctly higher in the vicinity of the singularities, the errors of the corrected approximation based upon p_1^y have the same order near the singularity and far from it. Also, the errors of the corrected approximation based upon p_2^y are much reduced near the singularity.

In table 1 and fig. 9, there is a comparison of the maximal errors on the entire domain for varying M .

Here too we can see that the errors of the original approximation attain a maximum that is independent of M , while the maximal errors of the corrected approximations decrease as M increases.

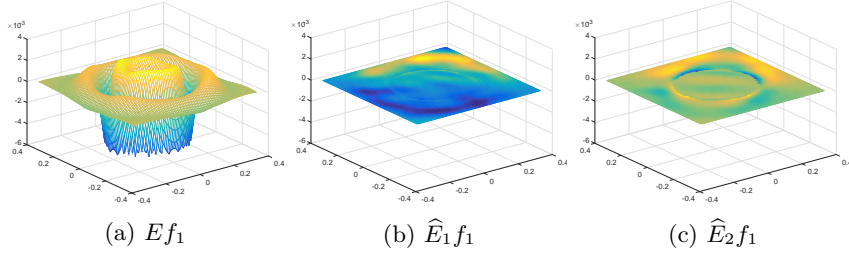


Figure 5: Comparison between the original MLS errors and the errors of our corrected approximations for f_1 .

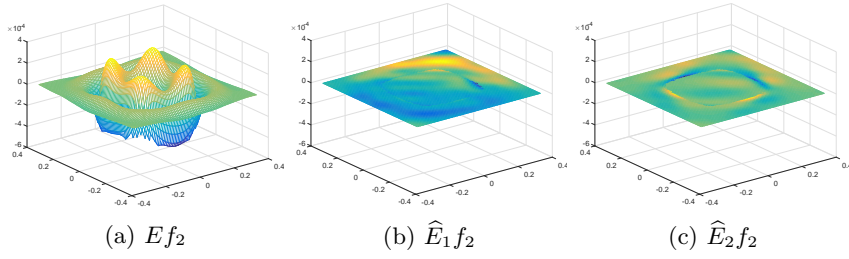


Figure 6: Comparison between the original MLS errors and the errors of our corrected approximations for f_2 .

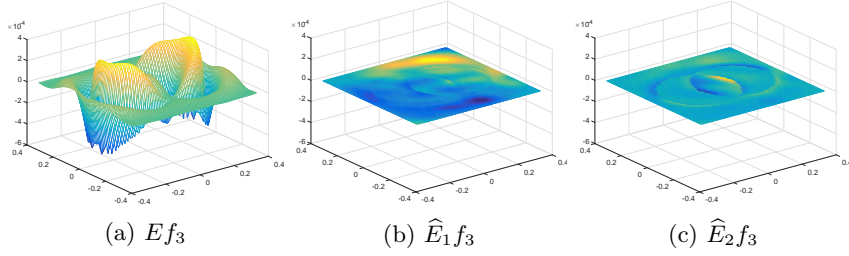


Figure 7: Comparison between the original MLS errors and the errors of our corrected approximations for f_3 .

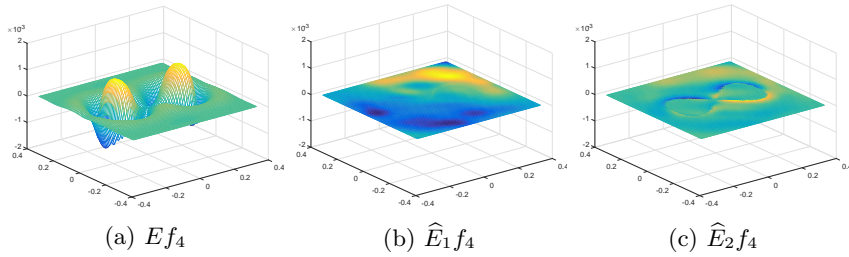


Figure 8: Comparison between the original MLS errors and the errors of our corrected approximations for f_4 .

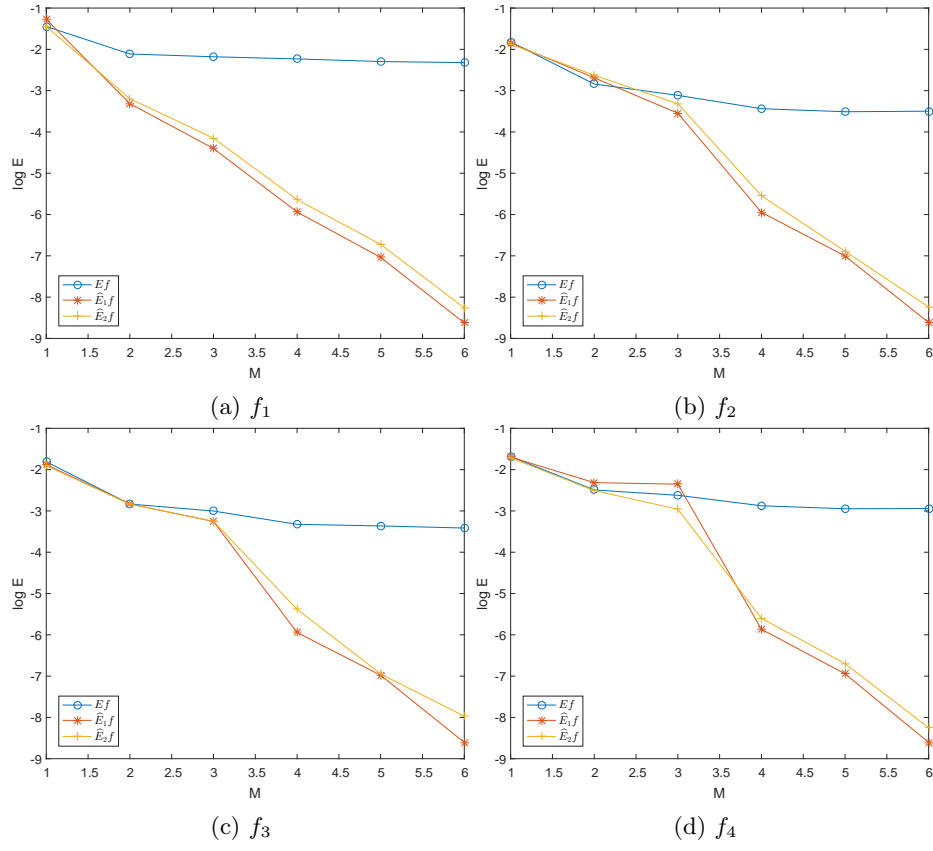


Figure 9: The errors of the original MLS approximation and the corrected approximations for varying M values for f_1 , f_2 , f_3 and f_4 .

Table 1: Comparison of the maximal errors for varying M .

M	$E(f_1)$	$\widehat{E}_1(f_1)$	$\widehat{E}_2(f_1)$	$E(f_2)$	$\widehat{E}_1(f_2)$	$\widehat{E}_2(f_2)$
1	$h^{0.99}$	$h^{0.87}$	$h^{0.99}$	$h^{1.32}$	$h^{1.34}$	$h^{1.36}$
2	$h^{1.45}$	$h^{2.27}$	$h^{2.19}$	$h^{2.06}$	$h^{1.95}$	$h^{1.91}$
3	$h^{1.49}$	$h^{3.02}$	$h^{2.84}$	$h^{2.26}$	$h^{2.58}$	$h^{2.41}$
4	$h^{1.53}$	$h^{4.07}$	$h^{3.86}$	$h^{2.50}$	$h^{4.32}$	$h^{4.03}$
5	$h^{1.57}$	$h^{4.82}$	$h^{4.61}$	$h^{2.55}$	$h^{5.09}$	$h^{5.01}$
6	$h^{1.59}$	$h^{5.91}$	$h^{5.66}$	$h^{2.54}$	$h^{6.26}$	$h^{5.99}$

M	$E(f_3)$	$\widehat{E}_1(f_3)$	$\widehat{E}_2(f_3)$	$E(f_4)$	$\widehat{E}_1(f_4)$	$\widehat{E}_2(f_4)$
1	$h^{1.28}$	$h^{1.33}$	$h^{1.35}$	$h^{1.18}$	$h^{1.19}$	$h^{1.20}$
2	$h^{2.00}$	$h^{2.00}$	$h^{2.00}$	$h^{1.74}$	$h^{1.62}$	$h^{1.75}$
3	$h^{2.12}$	$h^{2.30}$	$h^{2.30}$	$h^{1.83}$	$h^{1.64}$	$h^{2.07}$
4	$h^{2.35}$	$h^{4.21}$	$h^{3.81}$	$h^{2.01}$	$h^{4.10}$	$h^{3.92}$
5	$h^{2.38}$	$h^{4.94}$	$h^{4.92}$	$h^{2.06}$	$h^{4.86}$	$h^{4.68}$
6	$h^{2.42}$	$h^{6.09}$	$h^{5.64}$	$h^{2.06}$	$h^{6.02}$	$h^{5.77}$

In fig. 10, we drew the curves $\{r_k = 0\}$ on which the function f_k has singularities. We also drew the curves $\{p_1^y(y) = 0\}$ and $\{p_2^y(y) = 0\}$, for each r_k , these are our approximations of the curve $\{r_k = 0\}$. We have used the MATLAB `contour` command to draw these curves.

In these figures we see that the curves $\{r_k = 0\}$ are indistinguishable from their approximations based upon p_1^y and p_2^y .

In fig. 11, we outline the Hausdorff distance between the singularity curve $\{r_1 = 0\}$ and our approximations of this curve, $\{p_1^y = 0\}$ and $\{p_2^y = 0\}$ for varying values of fill-distance h . Note that as we took varying h values we did not change the number of data points, but only the size of the region in which we test the procedure.

In fig. 12, we compare the original approximation errors to the corrected approximations errors for varying h values.

Here also it is clear that the corrected approximation errors are much more affected by the decreasing h than the original approximation errors.

We also tested our partitioning with respect to the sign of r_k algorithm (see section 4) on our data set for $1 \leq k \leq 4$. In figs. 13 to 16 we show the steps of the partitioning algorithm.

Note how well the algorithm succeeds in partitioning the data points without any knowledge of the function r .

References

- [1] F. Arandiga, A. Cohen, R. Donat, and N. Dyn. Interpolation and approximation of piecewise smooth functions. *SIAM Journal on Numerical Analysis*, 43(1):41–57, 2005.

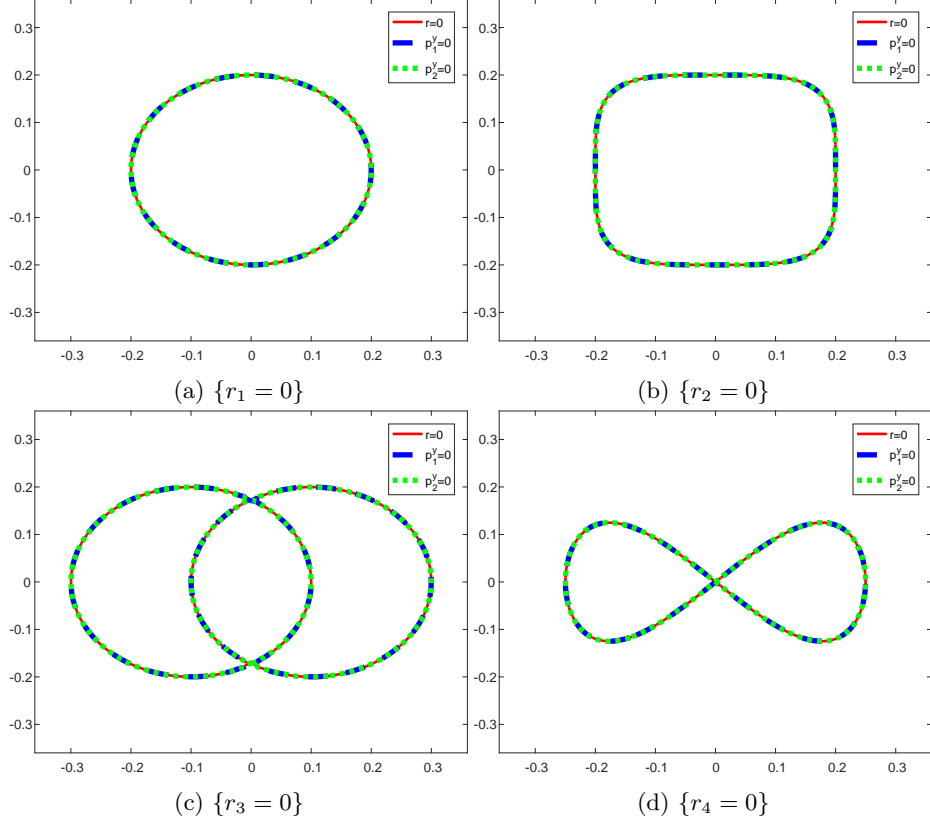


Figure 10: Comparison between the curves $\{r_k = 0\}$ (red) and their approximations $\{p_1^y = 0\}$ (blue dashed) and $\{p_2^y = 0\}$ (green dotted) for $k = 1, \dots, 4$.

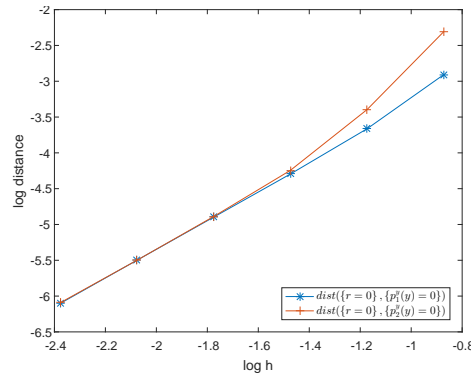


Figure 11: The Hausdorff distance between the curve $\{r_1 = 0\}$ and our approximations of this curve $\{p_1^y = 0\}$ and $\{p_2^y = 0\}$ for varying values of h .

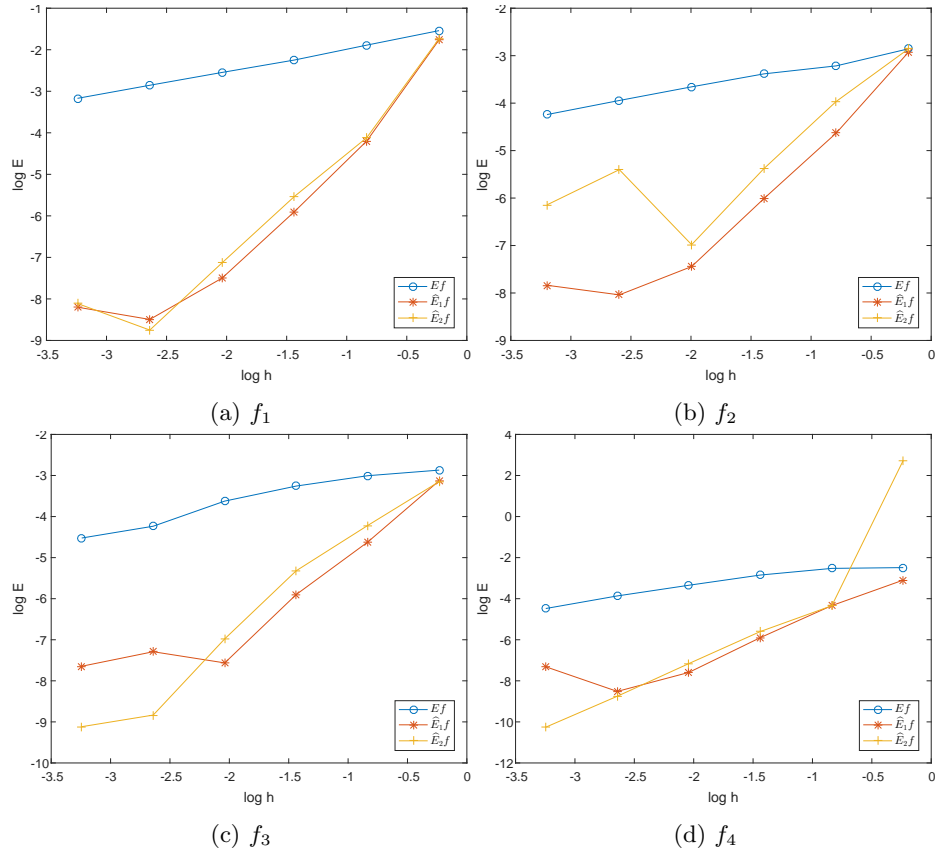


Figure 12: The errors of the original MLS approximation and the corrected approximations for varying h values for f_1 , f_2 , f_3 and f_4 .

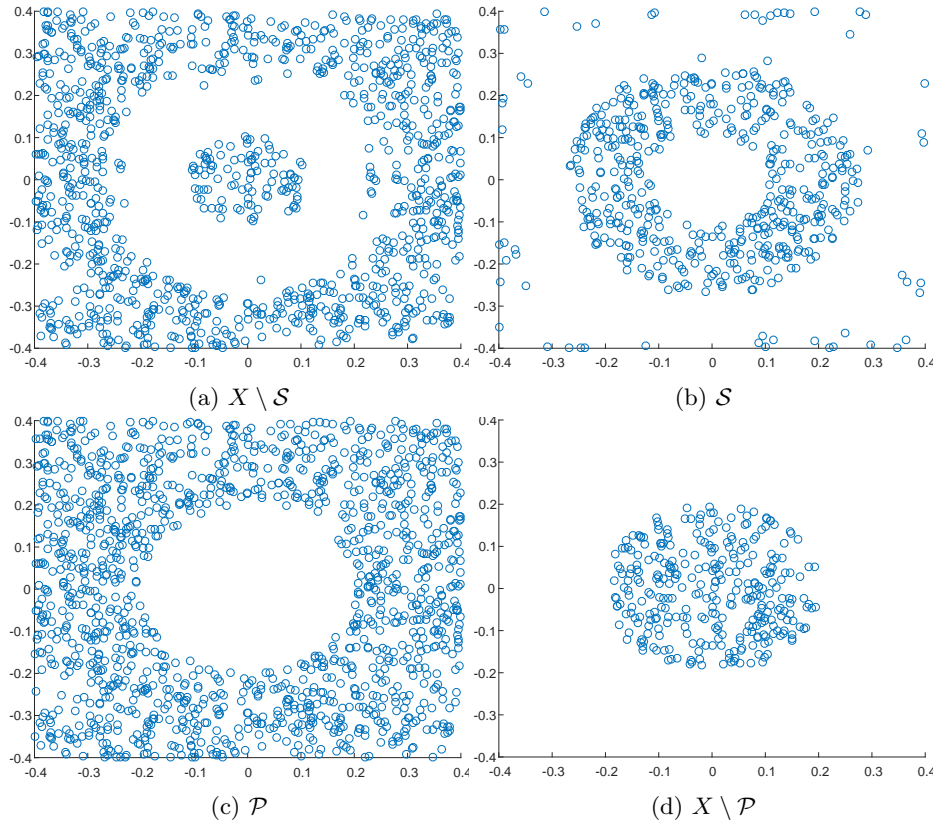


Figure 13: The results of the partitioning with respect to the signs of r_1 .

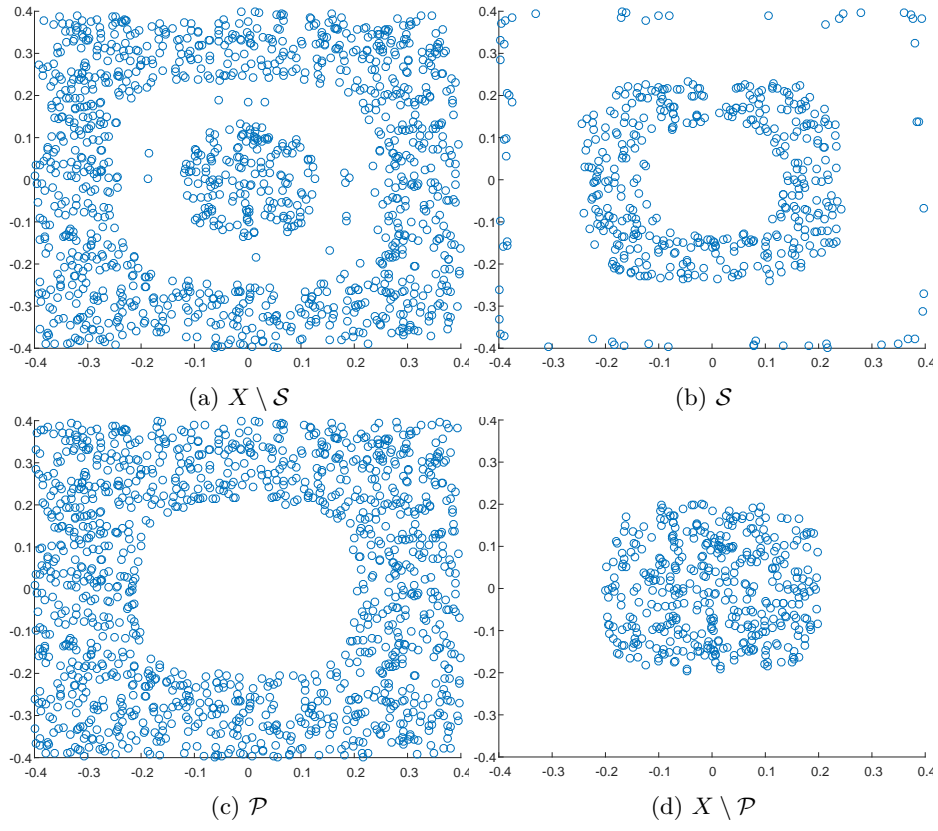


Figure 14: The results of the partitioning with respect to the signs of r_2 .

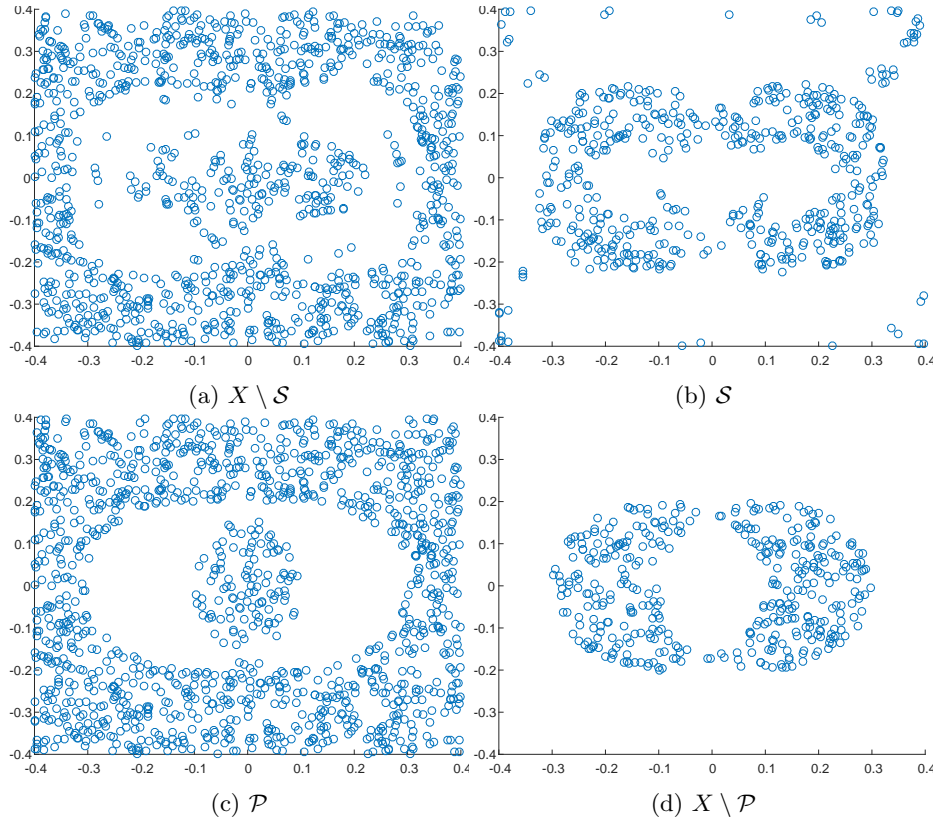


Figure 15: The results of the partitioning with respect to the signs of r_3 .

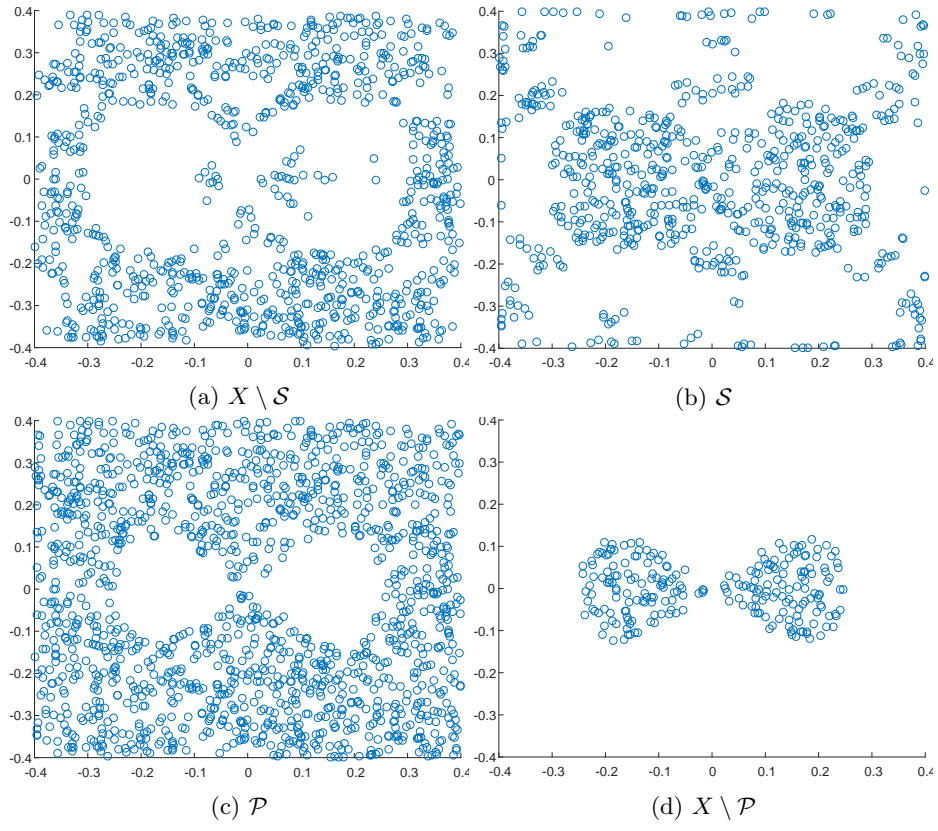


Figure 16: The results of the partitioning with respect to the signs of r_4 .

- [2] R. Archibald, A. Gelb, R. Saxena, and D. Xiu. Discontinuity detection in multivariate space for stochastic simulations. *Journal of Computational Physics*, 228(7):2676–2689, 2009.
- [3] R. Archibald, A. Gelb, and J. Yoon. Determining the locations and discontinuities in the derivatives of functions. *Applied Numerical Mathematics*, 58(5):577–592, 2008.
- [4] D. Batenkov. Complete algebraic reconstruction of piecewise-smooth functions from fourier data. *arXiv preprint arXiv:1211.0680*, 2012.
- [5] D. Batenkov, N. Sarig, and Y. Yomdin. Algebraic reconstruction of piecewise-smooth functions from integral measurements. *arXiv preprint arXiv:1103.3969*, 2011.
- [6] D. Batenkov and Y. Yomdin. Algebraic fourier reconstruction of piecewise smooth functions. *Mathematics of Computation*, 81(277):277–318, 2012.
- [7] L. Bos and K. Salkauskas. Moving least-squares are backus-gilbert optimal. *Journal of Approximation Theory*, 59(3):267–275, 1989.
- [8] A. Harten. ENO schemes with subcell resolution. *Journal of Computational Physics*, 83(1):148 – 184, 1989.
- [9] D. Levin. The approximation power of moving least-squares. *Mathematics of Computation of the American Mathematical Society*, 67(224):1517–1531, 1998.
- [10] Y. Lipman and D. Levin. Approximating piecewise-smooth functions. *IMA Journal of Numerical Analysis*, 30(4):1159–1183, 2010.
- [11] C. Markakis and L. Barack. High-order difference and pseudospectral methods for discontinuous problems. *arXiv preprint arXiv:1406.4865*, 2014.
- [12] P. Morse and K. Ingard. *Theoretical Acoustics*. International series in pure and applied physics. Princeton University Press, 1986.
- [13] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Applied Mathematical Sciences. Springer, 2003.
- [14] L. Plaskota and G. W. Wasilkowski. The power of adaptive algorithms for functions with singularities. *Journal of fixed point theory and applications*, 6(2):227–248, 2009.
- [15] L. Plaskota, G. W. Wasilkowski, and Y. Zhao. An adaptive algorithm for weighted approximation of singular functions over \mathbb{R} . *SIAM Journal on Numerical Analysis*, 51(3):1470–1493, 2013.
- [16] H. Wendland. *Scattered data approximation*, volume 17. Cambridge university press, 2004.